

SOURCE CODE GENERATION SYSTEM

Patent number: JP2003058370
 Publication date: 2003-02-28
 Inventor: YOSHIDA MASAMITSU; HASHIMOTO TOSHIKI
 Applicant: ALPHA:KK;; NIPPON YUNISHISU KK
 Classification:
 - international: G06F9/44
 - european:
 Application number: JP20010244396 20010810
 Priority number(s):

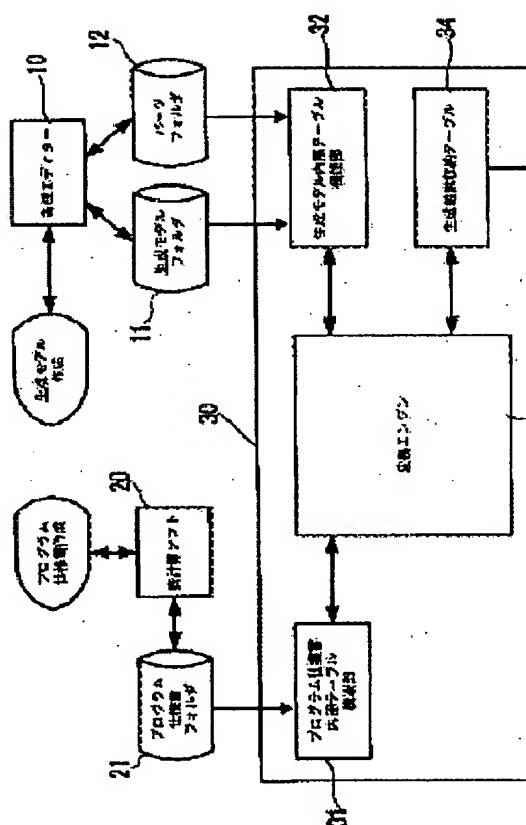
22390 U.S. PTO
 10/761207



Abstract of JP2003058370

PROBLEM TO BE SOLVED: To easily create a target source code without depending on a specific development language and specific items.

SOLUTION: A source code generation system is a generation model to be standard by every processing pattern and is provided with a generation model storage means for storing a generation model in a text form in which a control sentence to specify customizing conditions of the generation model is incorporated in a source code of the generation model, a program specification storage means for storing program specifications to be provided by spreadsheet software and that specifies the customizing conditions and a source code generation tool that enables selection of the generation model stored in the generation model storage means, enables selection of the program specification stored in the program specification storage means and generates the target source code based on the selected generation model and the program specification.



THIS PAGE BLANK (USPTO)

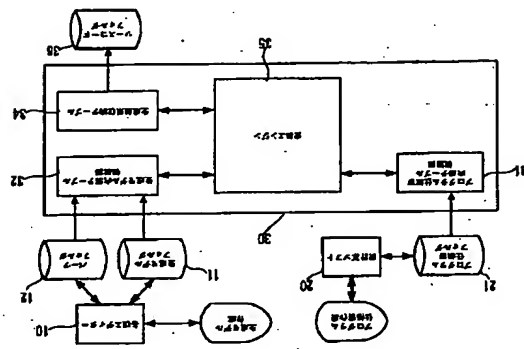
(51) Int. Cl.	識別記号	F I	チャート	(参考)
G06F 9/44		G06F 9/06	620 A	58076
			620 K	
審査請求 有 請求項の数 9 O L (全17頁)				
(21) 出願 号	特開2001-244396 (P 2001-244396)	(71) 出願人	501319036	
(22) 出願日	平成13年8月10日 (2001.8.10)	株式会社アルプアー		
		大阪府大阪市西区北堀江1丁目11番5号		
		(71) 出願人	591030237	
		日本ユニシス株式会社		
		東京都港区新坂2丁目17番51号		
		(72) 発明者	奥田 政光	
		大阪府大阪市西区北堀江1丁目11番5号		
		(74) 代理人	100059959	
		株式会社アルプアー内		
		弁護士 中村 益 (外9名)		

(54) 【発明の名称】 ソースコード生成システム

(57) 【要約】

【解決手段】 ソースコード生成システムは、処理パターン毎に標準となる生成モデルであって、該生成モデルのカスタマイズ条件を指定する制御文を生成モデルのソースコードに組み込んだテキスト形式の生成モデルを収納するための生成モデル収納手段と、該生成モデルによって与えられ且つカスタマイズ条件を指定するプログラム仕様書を受納するプログラム仕様書受納手段と、生成モデル収納手段に収納された生成モデルを選択できるようにし、プログラム仕様書受納手段に収納されたプログラム仕様書に基づいて目的のソースコードを生成するソースコード生成手段とを備える。

【効果】 特定の開発言語や特定項目に依存せず、簡単に目的のソースコードを作成することができる。



最終頁に続く

機能させるためのプログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、ソースコード生成システムに関するものである。

【0002】

【従来の技術】 ソフトウェア開発においては、短期間で高品質なソフトウェアを開発するという命題に対して様々なアプローチがなされてきた。その一つのアプローチとして、プログラム仕様からソースコードを自動生成するシステムの開発が行われている。

【0003】

この種のソースコード生成システムの従来の例としては、カスタマイズ可能な形で用意された標準プログラムパターンに対して、外部からカスタマイズパラメータを与え、その指示に従ってプログラムを自動生成するシステムがあり、これらシステムは、例えば、特開平4-140832号公報、特開平4-142622号公報等に開示されている。

【0004】

【発明が解決しようとする課題】 しかしながら、これら従来のソースコード自動生成システムには、次のような問題点があった。

(1) 特定の開発言語や特定項目 (例えば、データ項目) に特化したカスタマイズ形式および情報を用いていることにより、カスタマイズの対象となる標準プログラムモデルの言語 (例えば、COBOL、C言語、JAVA (登録商標)) の構文および特性に依存することになり、言語毎に対応したコード生成ツールを開発しなければならぬ。また、同一のプログラム仕様を別の言語で開発したい場合、その言語に対応したカスタマイズパラメータを新たに作成する必要があるが、プログラム仕様の活用を繰り返して利用できない。

(2) カスタマイズパラメータのみではプログラム仕様書としての直観性および体系的な与え難いものである。すなわち、カスタマイズパラメータは、自身がプログラム仕様の位置付けにあるが、パラメータの羅列はプログラム仕様書として直観性および体系的な与え難い。カスタマイズパラメータを編集して、プログラム仕様書に似たカスタマイズ実装情報書出力する機能をコード生成ツールに組み込むことも行われているが、ツールに依存した画一的な内容であった。

(3) カスタマイズパラメータの正確な入力为保证する仕組みに柔軟性や自由度が制限されている。何故ならば、ファイルのレコード項目名など共通項目の大入力、指定されたカスタマイズパラメータの正当性および妥当性検査機構の組込み、カスタマイズパラメータの再利用を実現するために、ソースコード生成ツール自身にその仕組みを作り込んでいくからである。

【0005】 本発明の目的は、前述したような従来の問題点を解消しようとするソースコード生成システムを提案することである。

【発明の構成】

【特許請求の範囲】

【請求項1】 プログラム仕様からソースコードを自動生成するソースコード生成システムにおいて、処理パターン毎に標準となる生成モデルであって、該生成モデルのカスタマイズ条件を指定する制御文を生成モデルのソースコードに組み込んだテキスト形式の生成モデルを収納するための生成モデル収納手段と、該生成モデルによって与えられ且つカスタマイズ条件を指定するプログラム仕様書を受納するプログラム仕様書受納手段と、前記生成モデル収納手段に収納された生成モデルを選択できるようにし、前記プログラム仕様書受納手段に収納されたプログラム仕様書に基づいて目的のソースコードを生成するソースコード生成手段とを備えることを特徴とするソースコード生成システム。

【請求項2】 前記生成モデルにおける制御文は、生成するソースコードが目的とする言語とは無関係である請求項1に記載のソースコード生成システム。

【請求項3】 前記ソースコード生成手段は、メニュー画面を介してプログラム仕様書および生成モデルを選択できるようにする選択手段と、選択されたプログラム仕様書とプログラム仕様に照準するためのプログラム仕様書内部テーブル構造部と、選択された生成モデルを内部テーブルに照準するための生成モデル内部テーブル構造部と、プログラム仕様書内部テーブル構造部および生成モデル内部テーブル構造部によって展開された内部テーブルに基づいて目的のソースコードを自動生成する変換エンジンとを含む請求項1または2に記載のソースコード生成システム。

【請求項4】 処理パターン別に実行プログラムを作成し、該作成した実行プログラムをベースに、制御文を組み込むことにより生成モデルを作成し、該作成した生成モデルを前記生成モデル収納手段に収納するようにするエディタ手段をさらに備える請求項1または2または3に記載のソースコード生成システム。

【請求項5】 前記プログラム仕様書を与えるための表計算ソフトをさらに備える請求項1から4のうちのいずれか1項に記載のソースコード生成システム。

【請求項6】 前記表計算ソフトは、エクセルである請求項5に記載のソースコード生成システム。

【請求項7】 前記ソースコード生成手段によって生成されたソースコードを収納するためのソースコード収納手段をさらに備える請求項1から6のうちのいずれか1項に記載のソースコード生成システム。

【請求項8】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを記録したコンピュータ読み取り可能な記録媒体。

【請求項9】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項10】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項11】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項12】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項13】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項14】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項15】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項16】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項17】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項18】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項19】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項20】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項21】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項22】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項23】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項24】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項25】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項26】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項27】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項28】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項29】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項30】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項31】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項32】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項33】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

【請求項34】 請求項1から7のうちのいずれか1項に記載のソースコード生成システムとしてコンピュータを機能させるためのプログラムを作成するプログラム。

供することである。

【0006】

【発明を解決するための手段】本発明によれば、プログラム仕組からソースコードを自動生成するソースコード生成システムにおいて、処理バターン毎に標準となる生成モデルであつて、該生成モデルのカスタマイズ条件を指定する制御手段を該生成モデルのソースコードに組み込んだテキスト形式の生成モデルを収納するための生成モデル収納手段と、該計算ソフトによって与えられ且つカスタマイズ条件を指定するプログラム仕組書を収納するプログラム仕組書収納手段と、前記生成モデル収納手段に収納された生成モデルを選択できるようにし、前記プログラム仕組書収納手段に収納されたプログラム仕組書を選択できるようにし、該選択された生成モデルおよびプログラム仕組書に基づいて目的のソースコードを生成するソースコード生成ツールとを備えることを特徴とする。

【0007】本発明の一つの実施形態によれば、前記生成モデルにおける制御文は、生成するソースコードが目的とする言語とは無関係である。

【0008】本発明のさらに別の実施形態によれば、前記ソースコード生成ツールは、メニュー画面を介してプログラム仕組書および生成モデルを選択できるようにする選択手段と、選択されたプログラム仕組書を内部テーブルに展開するためのプログラム仕組書内部テーブル機能部と、選択された生成モデルを内部テーブルに展開するための生成モデル内部テーブル機能部と、プログラム仕組書内部テーブル機能部および生成モデル内部テーブル機能部によって展開された内部テーブルに基づいて目的のソースコードを自動生成する変換エンジンとを含む。

【0009】本発明のさらに別の実施形態によれば、処理バターン別に該プログラムを作成し、該作成したプログラムをベースに、制御文を組み込むことにより生成モデルを作成し、該作成した生成モデルを前記生成モデル収納手段に収納させるようにするエディタ手段をさらに備える。

【0010】本発明のさらに別の実施形態によれば、前記プログラム仕組書を与えるための設計ソフトをさらに備える。

【0011】本発明の別の実施形態によれば、前記設計ソフトは、エディタである。

【0012】本発明のさらに別の実施形態によれば、前記ソースコード生成ツールによって生成されたソースコードを収納するためのソースコード収納手段をさらに備える。

【0013】本発明の別の観点によれば、前述のようなソースコード生成システムとしてコンピュータを機能させるためのプログラムを記録したコンピュータ読み取り可能な記録媒体が提供される。

【0014】本発明のさらに別の観点によれば、前述のようなソースコード生成システムとしてコンピュータを機能させるためのプログラムが提供される。

【0016】

【発明の実施形態】次に、附図面に基いて、本発明の実施形態および実施例について、本発明により詳細に説明する。

【0016】図1は、本発明の一実施例としてのソースコード生成システムの構成を、生成モデルの作成も含めてソースコード生成のための処理手順を説明し、図1に矢印で示す図であり、図2は、図1に関連して説明するための処理手順をフローチャートの形にまとめ示す図であり、図3は、図1のソースコード生成システムの構成を、入出力ファイルおよびコンピュータディスプレイに表示されるメニュー画面を中心として概略的に示す図である。

【0017】図1に示されるように、このシステムは、ソースコード生成において使用する生成モデルおよびパターンの作成するための各種エディタ10を備えており、これらエディタ10を介して作成された生成モデルおよびパターンは、それぞれを介して生成された生成モデル11およびパターン12に収納されるようになっている。また、このシステムは、ソースコード生成において使用するプログラム仕組書を作成するための設計ソフト20を、例えば、インテル社である、ここで、エディタ10とは、Microsoft Corporationが開発した設計ソフトMicrosoft Excel (登録商標) (以下、Excelと表記する) である。このような設計ソフト20を介して作成されたプログラム仕組書は、それぞれ対応するプログラム仕組書フォルダ21に収納される。さらに、このシステムは、ソースコード生成ツール30を備えており、このソースコード生成ツール30は、指定されたプログラム仕組書を内部テーブルに展開するプログラム仕組書内部テーブル機能部31と、指定された生成モデルを内部テーブルに展開する生成モデル内部テーブル機能部32と、生成されたソースコードを出力指示があるまで中間的に保存する生成結果格納テーブル34と、これら各機能部1〜32と相互動作してソースコードを生成するための動作を行う変換エンジン35とを備える。ソースコード生成ツール30によって生成されたソースコードは、ソースコードフォルダ36に収納される。

【0018】次に、図1のシステムにおいて、ソースコードを生成するための処理手順の全体の流れについて、図2のフローチャートを参照しつつ概略的に説明する。【0019】この実施例のシステムは、ソースコードの自動生成の前段階としての生成モデルの作成を行えるものであり、まず、このシステムを使用してモデル作成

業を行う。図2のフローチャートにおいて、ステップS1において、生成モデルの選択を行う。これは、本システムで作成するソースコードを処理バターン別に分類し、何種類の生成モデルを作成するかを決定する。これらの生成モデルの例としては、メニュー画面、条件指定画面、一覧表示画面、明細表示画面 (集計)、一覧更新画面、明細更新画面 (集計)、一覧更新画面、集計更新画面、種々のバッチ処理、データベース生成スクリプト、ファイルデータ定義項目、データベース定義項目等が挙げられる。生成モデルは、プログラムだけに限定されず、テキスト形式のソースコードで表現されるものであればよい。

【0020】次に、ステップS2において、生成モデルの選択プログラム (ソースコード) の作成を行う。ここでは、ステップS1において決定した処理バターン別に、実際のプログラムを作成し、テキストを基にプログラムを完成させる。ここで作成するコードはモデルとして使用するためコード規約、コメント等は全て正しく作成する必要がある。このステップS2と並行して、ステップS3において、本システムで必要とする共通機能部を作成する。例えば、ステップS4において、ソースコードのモデル化を行う。ここでは、ステップS2で作成した生成モデルの実際のプログラムをベースに、プログラム仕組書の条件を反映できるよう「制御コード (制御文)」を組み込む。このステップS4と並行して、ステップS5において、当該生成モデルのプログラム仕組書を作成するソフト20を、例えば、Excelベースで作成する。以上のステップS1からステップS5の作業は、本システムの準備段階であるが、開発する本システムの完成レベルを決定づける重要な作業である。

【0021】次に、こうして準備された本システムにおけるソースコード作成作業の概要について説明する。

【0022】まず、ステップS6において生成モデルの選択を行う。ここでは、作成すべきソースコード (プログラム) などの生成モデルに該当するかを検討する。この場合において、生成モデルの追加および変更が必要になることもある。

【0023】次に、ステップS7において、プログラム仕組書を作成するため、作成するプログラムの選択項目及び条件を記述する。この場合において、プログラム仕組書の形式変更が必要になることもある。

【0024】それから、ステップS8において、ステップS6において選択したベースとなる生成モデルとステップS7において作成したプログラム仕組書とよりソースコードを自動生成する。

【0025】以上のステップS6からステップS8の作業は、プログラム単位の作業となる。そして、最後に、ステップS9において、システムテストを行う。

【0026】以上が本システムにおいて行われる作業および動作の概要であるが、以下、本システムの各部の詳細構成および動作について、項分けて説明していく。

入出力ファイル

図1および図3に示されるプログラム仕組書フォルダ21は、Excelベースで作成したプログラム仕組書を格納するフォルダである。ここでは、1つのプログラム仕組書は1つのプログラム、複数のソースコードで記述している。このフォルダ21は、複数のプログラムの仕組書を格納する。

【0027】図1および図3に示される生成モデルフォルダ21は、テキスト形式で作成した生成モデルを格納するフォルダである。ここでは、1つの生成モデルのソースコードをテキスト形式で記述している。このフォルダ21は、複数のパターンの生成モデルを格納する。

【0028】図1および図3に示されるパターンフォルダ12は、生成モデルを部品化してテキスト形式で作成したパターンを格納するフォルダである。ここでは、1つのパターン (Component) 単位のソースコードをテキスト形式で記述している。このフォルダ12は、複数のパターンを格納する。

【0029】図3に示される初期設定ファイル13は、ソースコード生成ツール30に対するカスタマイズ情報やパラメータ形式で指定するプログラムである。ここでは、各フォルダのパス名、制御文種類、ログ要求等を定義/保存する。

【0030】図1および図3に示されるソースコードフォルダ36は、テキスト形式で生成されたソースコードを格納するフォルダである。ここでは、生成結果のソースコードがテキストファイルで作成される。このフォルダ36は、複数の生成結果を格納する。

【0031】図3に示されるログファイル37は、実行経過をテキスト形式で記録するファイルである。

メニュー画面

前述したように、本システムの処理手順における図2のフローチャートのステップS8でのソースコード生成は、ステップS7でのプログラム仕組書の選択 (必要に応じてプログラム仕組書の編集)、ステップS6での生成モデルの選択 (必要に応じて生成モデルの編集) を経て、ソースコード生成を指示することによって行われるのであるが、当該処理を行うための、ソースコード生成ツール30のメニュー画面の一例を、図3において参照番号40を付して示している。

【0032】図3に示されるように、このメニュー画面40には、プログラム仕組書選択リスト41、生成モデル選択リスト42、機能別コマンドボタン43等が表示される。図6は、このようなメニュー画面40の具体例を示している。以下、図6のメニュー画面に提示されている機能ボタンについて説明する。

機能ボタン

(1) 設計書選択: プログラム仕組書を選択するリストボックスを表示する。

(2) 設計書入力: プログラム仕組書を編集するExcel

を開く。

(3) 設計者ロード: プログラム仕様書をコード生成ツールの内部テーブルにロードする。

(4) 生成モデル選択: 生成モデルを選択するリストボックスを表示する。

(5) 生成モデル入力: 生成モデルを選択するため指定されたエディタを開く。エディタは、Microsoft(R)ワードパッド、WIPES、秀丸エディタ、... 等指定でき、初期設定ファイルで設定する。

(6) コード生成: 選択されたプログラム仕様書と生成モデルをベースにソースコード生成処理を行う。

(7) 逆読生成: パラメータファイルの指示情報を元に逆読的にソースコード生成処理を行う。ソースコード生成を逆読的に「プログラム仕様書」と「モデル名」を対応したパラメータファイルにバッチファイル(テキスト形式)として生成モデルフォルダに登録する。

プログラム仕様書

図7は、プログラム仕様書の構成を概念的に示しており、図8は、プログラム仕様書一般形式に示している。これら図に示されるように、プログラム仕様書は、プログラム単位にExcelの1ブックとして定義する。仕様の各ページは、ブック内の各シートに対応する。[0033] このプログラム仕様書の仕様としては次の点が挙げられる。

(1) 設計者から見た場合
(a) カスタマイズパラメータ以外の情報も記載させてピジュアルであり(図解、断接、カラー、フォント、フォントサイズ)、一般的な仕様書としての体裁が保てる。

(b) 一般的に使い慣れたソフトであり定義し易い。
(c) 印刷、複写、保存、変換等の操作が簡単にできる。

(d) Excelマクロ/VBAにより仕様書を定義する際のサポート機能を容易に実現できる。

(2) 生成ツールから見た場合

(a) セル単位の情報(シート、行、列)が参照できる。

(c) セル単位の情報(シート、行、列)は内部テーブル(3次元)として展開し易い。

(d) 複数シートに分割して定義したパラメータを行単如に紐付けし、1行の項目属性として参照/設定できる。

[0034] プログラム仕様書とセル値参照の関連
プログラム仕様上のセル値は「単要素型」と「明細行型」の2種類で参照できる。

(1) 「単要素型」... シートID、列ID、行番号を指定する方法

図7のセルA、セルB、セルCを参照する場合に用いる。

(2) 「明細行型」... 行番号を省略して、シートID

D. 列IDを指定する方法

行番号はDoループの現在行(Doループ制御処理の中で現在処理対象となっている行番号)が決定される。図7のセルA、セルB、セルCをDoループ制御処理の中で参照する場合に用いる。

[0035] 列ID、行番号は次のような形式により直感的でわかりやすい名前を指定することができる。

例1: \$XSheet!\$A\$1 (シートIDの識別文字)

例2: \$XSheet!\$A\$1 (シートIDの識別文字)

例3: \$XSheet!\$A\$1 (シートIDの識別文字)

例4: \$XSheet!\$A\$1 (シートIDの識別文字)

例5: \$XSheet!\$A\$1 (シートIDの識別文字)

例6: \$XSheet!\$A\$1 (シートIDの識別文字)

例7: \$XSheet!\$A\$1 (シートIDの識別文字)

例8: \$XSheet!\$A\$1 (シートIDの識別文字)

例9: \$XSheet!\$A\$1 (シートIDの識別文字)

例10: \$XSheet!\$A\$1 (シートIDの識別文字)

例11: \$XSheet!\$A\$1 (シートIDの識別文字)

例12: \$XSheet!\$A\$1 (シートIDの識別文字)

例13: \$XSheet!\$A\$1 (シートIDの識別文字)

例14: \$XSheet!\$A\$1 (シートIDの識別文字)

例15: \$XSheet!\$A\$1 (シートIDの識別文字)

例16: \$XSheet!\$A\$1 (シートIDの識別文字)

例17: \$XSheet!\$A\$1 (シートIDの識別文字)

例18: \$XSheet!\$A\$1 (シートIDの識別文字)

例19: \$XSheet!\$A\$1 (シートIDの識別文字)

例20: \$XSheet!\$A\$1 (シートIDの識別文字)

例21: \$XSheet!\$A\$1 (シートIDの識別文字)

例22: \$XSheet!\$A\$1 (シートIDの識別文字)

をループ処理する。

\$... 識別文字 (初期設定ファイルで定義)

Do... 固有な名

[XSheet]... パラメータ

例2: \$Set[WriteDate=Date]... ワーク変数 (WriteDate)

[0041] 代表的な制御文の形式、機能を説明する。

■ループ制御文

\$Do, \$Loop, \$ExitDo, \$NextLoop制御文

\$Doから\$Loopの範囲を所定の回数分ループする制御文である。

\$Doから\$Loopの範囲に定義するソースコード、制御文に制限はない。

\$ExitDoによりループを中止することができる。

\$NextLoopにより次のループに移行することができる。

\$Loopはループの終端として必ず指定する。

\$Doから\$Loopの範囲に別の\$Do~\$Loopを指定(多重化)することができ。

形式①: 「当座シートの明細行数をループする」

\$Do[SheetID, Start, End, Step]

\$Sid... シートID (シートID以外不可)

Start... 開始位置を示す数値 (省略時は1、定数、変数、関数、計算式可)

End... 終了位置を示す数値 (省略時は1、定数、変数、関数、計算式可)

Step... インクリメント値 (省略時は1、定数、変数、関数、計算式可)

■条件判定制御文

\$Select, \$Case, \$EndSelect制御文

条件付きで実行を制御する制御文である。

\$Select

\$Case(条件式1)... 条件式1を満足する場合に実行するステートメントを定義する。

\$Case(条件式2)... 条件式2を満足する場合に実行するステートメントを定義する。

\$Case(条件式3)... 条件式3を満足する場合に実行するステートメントを定義する。

\$Case(Else)... 条件式1、2、3とも満足しない場合に実行するステートメントを定義する (省略可能)。

\$EndSelect

[0042] (2) ソースコード

制御文として認識されないテキスト行は全てソースコードとして認識される。ソースコードには任意の文字列と置換パラメータ、カラー定義、ワーク変数、文字定数、置換パラメータはプログラム仕様のカスタマイズパラメータ (シート変数)、関数、ワーク変数、文字定数、数値定数、及びこれらの要素を組み合わせた計算式を定義できる。カラー定義パラメータはTab関数により、生成されるソースコードの出力カラー (桁位置) を定義できる。

シートID

シート名

最大行数

最大列数

明細行の開始行番号

明細行の終了行番号

ループ時の現在行番号 (Doループ制御処理の中で現在処理対象となっている行番号)

行単位の管理情報 (2次元目: 行番号と対応)

連続先シート番号

連続先行番号

セル値

シート番号は読み込んだシート順に1~nの番号を付与する。

[0038] 生成モデル

生成モデルは、このモデルを構成する「ソースコード」と生成ツールに対する「制御文 (制御コード)」を組み合わせて、テキスト形式で定義したファイルである。

(1) 制御文 (制御コード)

制御文を識別する識別文字と機能を示す固有な名及びパラメータで定義する。30種類以上の制御文が用意されており、定義するカラーに制限はない。制御文の一般形式は次の通りである。

\$固有な名[パラメータ]

\$: 制御文の識別文字

固有な名: 制御文の固有な名

パラメータ: [...] で囲まれた制御文の固有な名

[0039] パラメータには、識別文字%で始まるシート変数 (Excelで指定されたプログラム仕様 (カスタマイズ情報) を示すセル値を参照するものである)、識別文字#で始まる関数 (本システムであらかじめ用意されているもので、文字列操作の関数、シート操作の関数、日付関連の関数などがある)、識別文字\$で始まるワーク変数 (中間結果保存など作業用変数)、文字定数、数値定数、四則演算子、比較演算子、論理演算子を組み合わせた計算式を指定できる。

[0040] 主な制御文としては、例えば、\$Do, \$Loop, \$ExitDo, \$NextLoop制御文等のようなループ制御文、例えば、\$Select, \$Case, \$EndSelect制御文等のような条件判定制御文、例えば、\$Call, \$Sub, \$EndSub制御文、その他、\$Label制御文 (ラベル設定)、\$Link/UnsetLink制御文 (行データ連結)、\$Write/EndWrite制御文 (シート修飾)、\$Copy制御文 (ページ複製)、\$Load制御文 (追加仕様の読み込み)、\$Save制御文 (メタデータ出力)、\$Break制御文 (生成結果出力)、\$Option制御文 (オプション指定)、\$Repeat制御文 (注釈) がある。

D. 列IDを指定する方法

行番号はDoループの現在行(Doループ制御処理の中で現在処理対象となっている行番号)が決定される。図7のセルA、セルB、セルCをDoループ制御処理の中で参照する場合に用いる。

[0035] 列ID、行番号は次のような形式により直感的でわかりやすい名前を指定することができる。

例1: \$XSheet!\$A\$1 (シートIDの識別文字)

例2: \$XSheet!\$A\$1 (シートIDの識別文字)

例3: \$XSheet!\$A\$1 (シートIDの識別文字)

例4: \$XSheet!\$A\$1 (シートIDの識別文字)

例5: \$XSheet!\$A\$1 (シートIDの識別文字)

例6: \$XSheet!\$A\$1 (シートIDの識別文字)

例7: \$XSheet!\$A\$1 (シートIDの識別文字)

例8: \$XSheet!\$A\$1 (シートIDの識別文字)

例9: \$XSheet!\$A\$1 (シートIDの識別文字)

例10: \$XSheet!\$A\$1 (シートIDの識別文字)

例11: \$XSheet!\$A\$1 (シートIDの識別文字)

例12: \$XSheet!\$A\$1 (シートIDの識別文字)

例13: \$XSheet!\$A\$1 (シートIDの識別文字)

例14: \$XSheet!\$A\$1 (シートIDの識別文字)

例15: \$XSheet!\$A\$1 (シートIDの識別文字)

例16: \$XSheet!\$A\$1 (シートIDの識別文字)

例17: \$XSheet!\$A\$1 (シートIDの識別文字)

例18: \$XSheet!\$A\$1 (シートIDの識別文字)

例19: \$XSheet!\$A\$1 (シートIDの識別文字)

例20: \$XSheet!\$A\$1 (シートIDの識別文字)

例21: \$XSheet!\$A\$1 (シートIDの識別文字)

例22: \$XSheet!\$A\$1 (シートIDの識別文字)

11
きる。Tab関数の定義形式は次のとおり、
[Tab(mn)] Tab関数は生成するソースコードのカラム
(半角文字として) 合わせを行う。Tab関数は任意の行
に値数値、記述できる。

変換後ソースコード
1カラム 20カラム 40カラム
AAAA BBBB CCCC

COBOLソースコードを例に置換パラメータ、カラム定義
パラメータの指定例を示す。

例1: PROGRAM-ID. [Tab(20)] [KsheetRPID
(4)].
[Tab(20)]はカラム定義パラメータで、20カラム目に
生成位置を合わせる。
[KsheetRPID(4)]は置換パラメータで、Ksheetの4行目
のRPID列のセル値で置き換える。

例2: \$Title[Ksheet]
\$D([項目名]) PICTURE [ダイナミク]。
\$Loop
\$endwith. . . Ksheetの明細行をループし、列名 [項目
名]に直列で設定されている行を対象に列名 [項目名]、
列名 [ダイナミク] の設定値を各々置換される。

[0043] (3) 生成モデルの特徴
本システムにおいて用いる生成モデルの特徴をまとめて
示すと次のとおりである。

①生成モデルを外部ファイルに定義し、生成ツールに与
えている。これにより複数の生成モデルを準備できる。
また生成モデルの作成、変更が容易である。
②モデルとなるソースコードに制御文を嵌せている。
③生成モデルのソースコードは下記の2点でのみ制約さ
れる。この結果、生成するソースコードは特定のプログ
ラム開発言語に限定する必要はない。
(プログラムのソースコード以外でも、パラメータ、JC
I等ソースコードであれば何でも生成できる。)

「生成モデルのソースコードに関する制約事項」
置換パラメータの定義形式
カラム定義パラメータの定義形式

置換パラメータはソースコード中に定義し、定義したパ
ラメータの計算結果により置換される置換記号である。
置換パラメータにはプログラムの仕様のカスタマイズパ
ラメータ (シート変数)、関数、ワーク変数、文字定
数、数値定数、及びこれらの要素を組み合わせた計算式
を定義することができる。カラム定義パラメータはソー
スコード中に定義し、生成されるソースコードの出力カ
ラム (桁位置) を指定する置換記号である。

④制御文は一般的なプログラム開発言語に収めている。
ルーチン構造 (Do)、条件分岐 (Select)、部品化 (Sc
op)、共通化 (Sub)、数値演算、文字列操作、ワー
ク変数等が定義できて高度なカスタマイズが可能であ
る。

mn. . . カラムを示す数値、変数、計算式を定義する。
例 変換前ソースコード AAAA[Tab(2
0)]BBBB[Tab(40)]CCCC

る。(プログラミミングに近い感覚でモデルを作成でき
る。)

[0044] 生成モデルを展開する内部テーブルの展開
図1に関して前述したように、ソースコード生成ツールの
3.0は、生成モデル内部テーブル機能部3.2において、
指定された生成モデルを以下の如く内部テーブルに展開
する。

・行単位の管理情報 (1次元目)
・ダイナミク (ソースコードまたは制御文の種類)
・階層レベル (\$D、\$Select、\$Sub階層の深さ)
・上方向リンク (関係する制御文の上方向リンク情報)
・下方向リンク (関係する制御文の下方向リンク情報)
・テキスト (ソースコードまたは制御文のテキスト内
容)
・制御文のパラメータ情報 (1) 内のパラメータを保
持)

・\$D/\$Selectの管理情報 (\$Dルーチン制御処理および
\$Selectの制御情報)
・\$Dルーチンのダイナミク (シート/シート列/ワーク変数)
・\$Dルーチンの開始行
・\$Dルーチンの現在行
・\$Dルーチンの終了行
・\$Dルーチンの行間隔
・シート番号
・列番号

・\$Select/Caseの判定結果
[0045] コード生成の動作原理
ソースコード生成の処理手順の概要については、図2の
フローチャートを参照して本システムの展開のための処
理手順と共に前述したのであるが、より分かり易くす
るため、図4にソースコード生成の処理手順のみをまと
めて示している。

[0046] この図4に示されるように、本システムに
おけるソースコード生成の動作は、概略的には次のよう
である。すなわち、Excelベースのプログラムの仕様と
生成モデルとなるソースコードとに基づいて、ソースコ
ード生成ツール3.0が、ソースコードに記述された制御
コード (制御文) を解釈し、また、プログラムの仕様書に
記述されたパラメータを解釈し、制御コードを実行する
ことにより、目的のソースコードが生成される。

[0047] 図5は、このようなソースコード生成動作
のより詳細なステップを示すフローチャートである。な
お、図5において、処理インデックス=生成モデル内部
テーブルの処理インデックスである。以下、この図5の
フローチャートに沿って、コード生成動作についてより
詳細に説明する。

[0048] 本システムにおけるソースコード生成は、
内部テーブルに展開したプログラムの仕様書および生成モ
デルをベースに処理されるもので、まず、ソースコード
生成ツール3.0は、図5のステップS8.1において、プ
ログラム仕様書内部テーブル機能部3.1がプログラムの仕
様書を内部テーブルに展開する。これは、選択されたプ
ログラム仕様書を内部テーブル (シートID、行番号、
列IDの3次元) に読み込むことにより行われる。次
いで、ステップS8.2において、生成モデル内部テー
ブル機能部3.2が生成モデルを内部テーブル (行単
位) に読み込むことにより行われる。すなわち、生成
モデルは、行単位で内部テーブルに展開され、基本的に
1行目から最終行に向け1行ずつ処理される。

[0049] 次に、ステップS8.3において、パーツの
抜きを行う。これは、ステップS8.2で生成された内部テ
ブルより\$Dop制御文 (パース抜き) を探索し、\$Dop
制御文のパラメータよりパーステーブル名を得て、パ
ーステーブル名を例にしたリンク付け

行 1 ソースコード1 0 0
2 ソースコード2 0 0
3 \$Select 4 0
4 \$Case[条件1] 7 3
5 ソースコード3 0 0
6 ソースコード4 0 0
7 \$Case[条件2] 10 3
8 ソースコード5 0 0
9 ソースコード6 0 0
10 \$Case[条件3] 13 3
11 ソースコード7 0 0
12 ソースコード8 0 0
13 \$Case[Else] 16 3
14 ソースコード9 0 0
15 ソースコード10 0 0
16 \$EndSelect 0 3
17 ソースコード11 0 0
18 ソースコード12 0 0

下方向リンク 上方向リンク

ソースコード12より該当するパースを、\$Dop制御文の
行位置に読み込むことにより行われる。そしてステッ
プS8.4において、生成モデル内部テーブル機能部3.2
が生成モデルを解析する。これは、ステップS8.2および
ステップS8.3で作成された内部テーブルをベースにソー
スコードと制御文を区分する。制御文の種類毎に区分す
る。制御文のパラメータ部を取り出すことにより行わ
れる。それから、ステップS8.5において、関係する制
御文をリンクする。これは、S8.4で解析された内部テ
ブルをベースに関連する制御文間を関係情報 (下方向
リンク、上方向リンク) でリンクすることにより行わ
れる。

関連する制御文: \$Dop/\$Loop、\$Select/\$Case/\$CaseIso
/\$EndSelect、\$Sub/\$EndSub
下方向のリンク: \$Dop/\$Loop、\$Select/\$Case/\$CaseIso
/\$EndSelect、\$Sub/\$EndSub
上方向のリンク: \$Loop/\$Dop、\$Case/\$Select、\$CaseIso
/\$EndSelect、\$EndSub/\$Sub

ここで、関係する制御文をリンクすることの意味は、
「元に戻る」、「次に進む」、「戻り飛ばす」等の処理
を効率よくすることにある。リンク付けの例を示す。

[0050] ステップS8.5において関係する制御文を
リンクし終えたらステップS8.6に進む。ステップS8.6
から処理インデックス3.5が処理する。まず、ステップS8.
6において、処理インデックスに1を設定し、ステップ
S8.7において、その処理インデックスが最終行より大
きい場合は、そこで処理を終了する。そうでない場合
は、ステップS8.8において、処理インデックスが示す
行が制御文であるか否かの判定を行い (生成モデル内部

13
14
ツアル2より該当するパースを、\$Dop制御文の
行位置に読み込むことにより行われる。そしてステッ
プS8.4において、生成モデル内部テーブル機能部3.2
が生成モデルを解析する。これは、ステップS8.2および
ステップS8.3で作成された内部テーブルをベースにソー
スコードと制御文を区分する。制御文の種類毎に区分す
る。制御文のパラメータ部を取り出すことにより行わ
れる。それから、ステップS8.5において、関係する制
御文をリンクする。これは、S8.4で解析された内部テ
ブルをベースに関連する制御文間を関係情報 (下方向
リンク、上方向リンク) でリンクすることにより行わ
れる。

関連する制御文: \$Dop/\$Loop、\$Select/\$Case/\$CaseIso
/\$EndSelect、\$Sub/\$EndSub
下方向のリンク: \$Dop/\$Loop、\$Select/\$Case/\$CaseIso
/\$EndSelect、\$Sub/\$EndSub
上方向のリンク: \$Loop/\$Dop、\$Case/\$Select、\$CaseIso
/\$EndSelect、\$EndSub/\$Sub

ここで、関係する制御文をリンクすることの意味は、
「元に戻る」、「次に進む」、「戻り飛ばす」等の処理
を効率よくすることにある。リンク付けの例を示す。

[0051] 処理インデックスが示す内部テーブルの最
当行が制御文の場合、ステップS8.9において、各制
御文固有の処理を行うのであるが、その例を示すと次の
ようである。

ソースコード12より該当するパースを、\$Dop制御文の
行位置に読み込むことにより行われる。そしてステッ
プS8.4において、生成モデル内部テーブル機能部3.2
が生成モデルを解析する。これは、ステップS8.2および
ステップS8.3で作成された内部テーブルをベースにソー
スコードと制御文を区分する。制御文の種類毎に区分す
る。制御文のパラメータ部を取り出すことにより行わ
れる。それから、ステップS8.5において、関係する制
御文をリンクする。これは、S8.4で解析された内部テ
ブルをベースに関連する制御文間を関係情報 (下方向
リンク、上方向リンク) でリンクすることにより行わ
れる。

関連する制御文: \$Dop/\$Loop、\$Select/\$Case/\$CaseIso
/\$EndSelect、\$Sub/\$EndSub
下方向のリンク: \$Dop/\$Loop、\$Select/\$Case/\$CaseIso
/\$EndSelect、\$Sub/\$EndSub
上方向のリンク: \$Loop/\$Dop、\$Case/\$Select、\$CaseIso
/\$EndSelect、\$EndSub/\$Sub

ここで、関係する制御文をリンクすることの意味は、
「元に戻る」、「次に進む」、「戻り飛ばす」等の処理
を効率よくすることにある。リンク付けの例を示す。

[0051] 処理インデックスが示す内部テーブルの最
当行が制御文の場合、ステップS8.9において、各制
御文固有の処理を行うのであるが、その例を示すと次の
ようである。

ソースコード12より該当するパースを、\$Dop制御文の
行位置に読み込むことにより行われる。そしてステッ
プS8.4において、生成モデル内部テーブル機能部3.2
が生成モデルを解析する。これは、ステップS8.2および
ステップS8.3で作成された内部テーブルをベースにソー
スコードと制御文を区分する。制御文の種類毎に区分す
る。制御文のパラメータ部を取り出すことにより行わ
れる。それから、ステップS8.5において、関係する制
御文をリンクする。これは、S8.4で解析された内部テ
ブルをベースに関連する制御文間を関係情報 (下方向
リンク、上方向リンク) でリンクすることにより行わ
れる。

19

【図2】図1に関連して説明する生成モデルの作成をも含めたソースコード生成のための処理手順を示すフロー図である。

【図3】図1のソースコード生成システムの構成をメニュー画面を中心として示す概略図である。

【図4】ソースコード生成の処理手順のみをまとめて示す図である。

【図5】ソースコード生成動作のより詳細なステップを示すフロー図である。

【図6】メニュー画面の具体例を示す図である。

【図7】プログラム仕様書の構成を概念的に示す図である。

る。

【図8】プログラム仕様書を一般形式にて示す図である。

【図9】図8のプログラム仕様書を展開した内部データ

【図10】図8のプログラム仕様書を展開した内部データのイメージを示す図である。

【図11】ソースコード生成の処理手順において使用されるイメージを示す図である。

【図12】図11の生成モデルの構成を表にまとめたもの。生成モデルの具体例を示す図である。

【図13】図11の生成モデルを展開した内部テーブルを示す図である。

のイメージを示す図である。

【図4】図1の生成モデルを展開した内部階層のイメージを示す図である。

【図5】本発明のシステムにて生成されたソースコードの具体例を示す図である。

【符号の説明】

10	各種エディター
11	生成モデルフォルダ
12	ヘッダフォルダ
13	初期設定ファイル
20	設計書ソフト
21	プログラム仕様書フォルダ
30	ソースコード生成ツール
31	プログラム仕様書内部ツール機能部
32	生成モデル内部ツール機能部
33	生成結果収納ツール
34	変換エンジン
35	ソースコードフォルダ
36	ログファイル
37	コマンド
40	メニュー画面
41	プログラム仕様書選択リスト
42	生成モデル選択リスト
43	機能別メニューボタン

のイメージを示す図である。

【図14】図11の生成モデルを展開した内部テーブルのイメージを示す図である。

【図15】本発明のシスデムにて生成されたソースコードの具体例を示す図である。

【符号の説明】

10 各種エディター

11 生成モデルフオルダ

12 パーグフオルダ

1.3 初期設定モデル

20 幸計算ノ77b

20 全三卷ノ一

21 ノロクナム任録會ノオルタ

30 ソースコード生成ツール

31 プログラム仕様書内部テーブル

3.2 生成モデル内部テーク

3.4 生成結果収納テーブル

35 変換エンジン

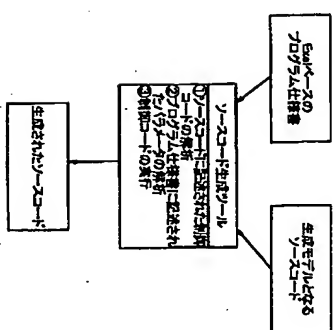
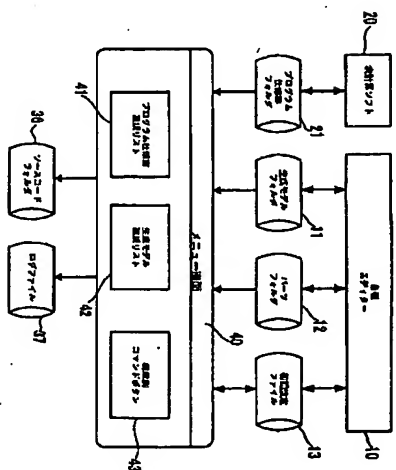
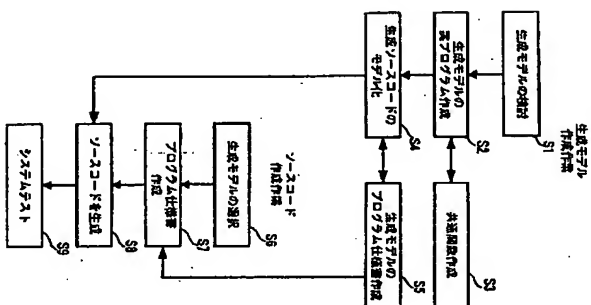
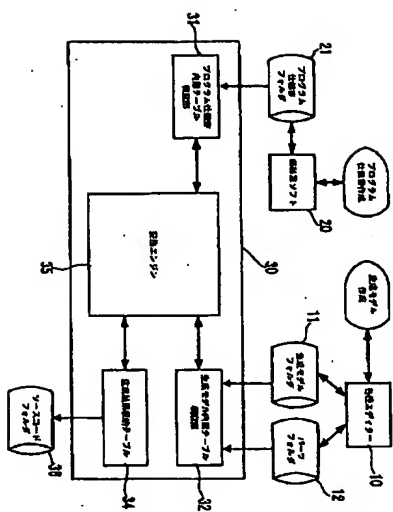
36 ソースコードフォルダ

37 047-11

[illegible]

40 × 25 一面面

4.1 プログラム仕様選択システム



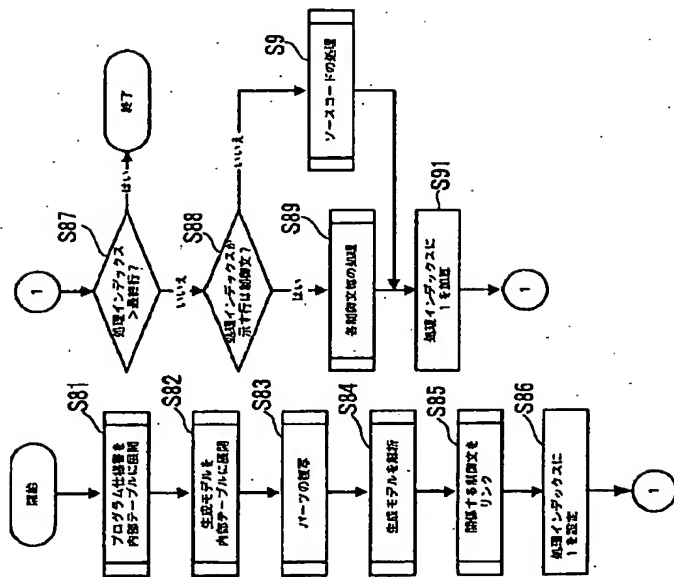
【6图】

[illegible]

100-138412-0 售 處 詳 見

[illegible]

【图5】



【例6】

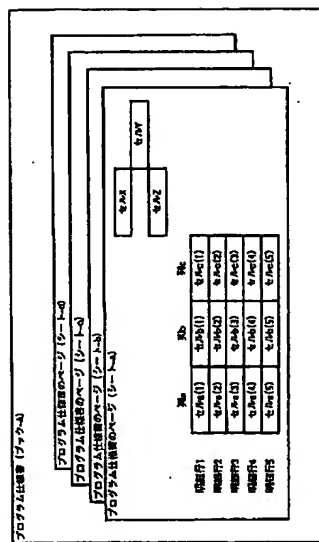
[圖 10]

[illegible]

【図8】

[illegible]

【図7】



【図1.1】

1	Option 0	1	Option 0
2	With 0	2	With 0
3	Case 0	3	Case 0
4	Case 0	4	Case 0
5	Case 0	5	Case 0
6	Case 0	6	Case 0
7	Case 0	7	Case 0
8	Case 0	8	Case 0
9	Case 0	9	Case 0
10	Case 0	10	Case 0
11	Case 0	11	Case 0
12	Case 0	12	Case 0
13	Case 0	13	Case 0
14	Case 0	14	Case 0
15	Case 0	15	Case 0
16	Case 0	16	Case 0
17	Case 0	17	Case 0
18	Case 0	18	Case 0
19	Case 0	19	Case 0
20	Case 0	20	Case 0
21	Case 0	21	Case 0
22	Case 0	22	Case 0

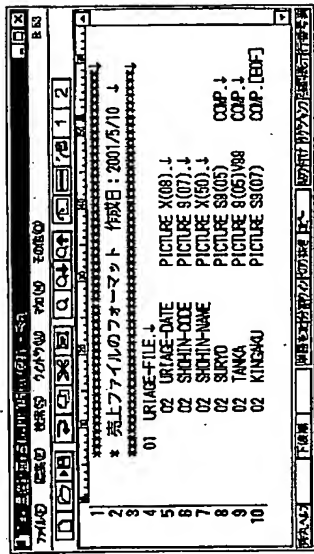
【図1.2】

行番号	種類	説明
1	Option 制御文	オプション (1) を指定して実行します。
2	With 制御文	With 文の範囲として "SheetID" を指定します。
3~5	Case 制御文	出力を指定します。
6, 8,	Case 制御文	そのまゝ出力されます。
9	Case 制御文	指定の Case (1) が実行される出力されます。
10	Do 制御文	シート SheetID の範囲 (5~11 行) をループします。列 Range のセル値が変更される行を指定し、以下 Loop の範囲をループします。
11	Select 制御文	Select/Case の指定をします。
12	Case 制御文	列 Range のセル値が変更で、且つ 列 Range のセル値が変更の場合は「条件文を満足する」として本行に制御を渡します。
13	Case 制御文	指定の Case (1) が実行される出力されます。
14	Case 制御文	列 Range のセル値が変更で、且つ 列 Range のセル値が変更の場合は「条件文を満足する」として本行に制御を渡します。
15	Case 制御文	指定の Case (1) が実行される出力されます。
16	Case 制御文	行 12, 14 の Case を「いずれも満足しない」場合、本行に制御を渡します。
17	Case 制御文	指定の Case (1) が実行される出力されます。
18	Case 制御文	Select/Case の終了をします。
19	Case 制御文	Do/Loop の終了をします。
20	Case 制御文	シート ID の指定で "SheetID" を指定します。
21	Case 制御文	列 Range の行目のセル値と文字列 "txt" を連結し、"SheetID.txt" をファイル名として生成結果を出力します。
22	Case 制御文	マクロ "マクロ" を実行しました。"マクロ" 画面に実行します。

【図1.3】

行	タイプ	レベル	上リンク	下リンク	テキスト	パラメータ	Do/Select の実行結果
1	Option	0	0	0	Option(1)	1	
2	With	0	0	0	With(SheetID)	SheetID	
3	Case	0	0	0	Case 0		
4	Case	0	0	0	Case 0		
5	Case	0	0	0	Case 0		
6	Case	0	0	0	Case 0		
7	Case	0	0	0	Case 0		
8	Case	0	0	0	Case 0		
9	Case	0	0	0	Case 0		
10	Do	1	0	10	Do(Option)	Option	有り
11	Select	1	0	11	Select		有り
12	Case	1	11	12	Case(Option and Usage)	Option and Usage	
13	Case	1	0	0	(Option) (Usage)		
14	Case	1	11	14	Case(Option and Usage)	Option and Usage	
15	Case	1	0	0	(Option) (Usage) [Tab2] PICTURE (Picture)		
16	Case	1	11	16	Case(Else)	Else	
17	Case	1	0	0	(Option) (Usage) [Tab2] PICTURE (Picture) [Tab		
18	EndSelect	1	11	0	EndSelect		
19	Loop	1	10	0	Loop		
20	EndWith	0	0	0	EndWith		
21	Break	0	0	0	Break(Picture() & ".txt")	Picture() & ".txt"	
22	Msg	0	0	0	Msg("コード生成を完了しました。")	"コード生成を完了しました。"	

【図15】



フロントページの続き

(72)発明者 橋本 俊朗

東京都港区赤坂2丁目17番51号 日本ユニ

シス株式会社内

Fターム(参考) 58076 AB10 DC00 DC02 DC04 DD00

DD02 DD04 DD05 DD06 DD08

DD10

THIS PAGE BLANK (USPTO)